**ТЕОРЕТИЧЕСКИЙ МАТЕРИАЛ.**

Чтобы **CPU** мог передать адрес какой-либо ячейки **RAM**,нужно записать ее адрес в один из регистров **CPU**. Максимальное число, которое в 16-тиразрядном процессоре можно записать в **16**-тиразрядный регистр, составляет всего **216 – 1 = 65535**, т.е. **64**Кбайт - 1. Следовательно, мы можем обращаться по номеру только к первым **64** Кб **RAM**. Для того чтобы с помощью **16**-тиразрядных чисел адресовать любой байт **RAM**, объемом в 1 Мбайт, в **CPU i8086** предусмотрена ***сегментная адресация памяти.*** Она реализуется с помощью специальных сегментных регистров **CPU**. При сегментной адресации обращение к ячейкам **RAM** осуществляется исключительно с помощью сегментов.

***Сегмент* – это логическая единица памяти с максимальным размером в 64 Кбайта (мин. размер – 16 байт), накладываемая на те или иные участки памяти.**

Шина адреса 20-тиразрядная, но нет ни одного 20-тиразрядного регистра, поэтому внутри CPU адрес ячейки памяти представлен двумя 16-битными словами: базовый (начальный) адрес сегмента и внутрисегментное смещение. Эти два слова, разделенные двоеточием, представляют собой ***логический адрес ячейки памяти***.

На размещение сегментов в пространстве в 1 Мбайт накладывается только одно ограничение – их базовый физический адрес должен быть кратен 16, т.е. его младшие 4 бита должны быть равны 0 (на границе параграфа). Нулевые биты можно не хранить, поэтому под этот адрес достаточно 16 бит. Начальный адрес сегмента без **4**-х младших битов, т.е. деленный на **16**, заносится в сегментный регистр и называется ***сегментным адресом.***

Для локализации конкретного байта в сегменте служит вторая компонента логического адреса – ***смещение***. Она является 16-битным целым, беззнаковым числом и показывает расстояние от начала сегмента до этого байта.

**Чтобы обратиться к нужной ячейке памяти, нужно сначала определить базовый адрес сегмента, в котором она находиться, а затем от начала этого сегмента переместиться на величину смещения.**

Смещение иначе называется ***относительным адресом***.

**ФИЗИЧЕСКИЙАДРЕС (PA) В РЕАЛЬНОМ РЕЖИМЕ ПОЛУЧАЕТСЯ СЛЕДУЮЩИМ ОБРАЗОМ:**

|  |
| --- |
| ***Сегментный адрес (Seg) умножается на 16(10h)*** *(в 16-ой системе это означает просто сдвиг всех цифр числа на одну позицию влево с добавлением 0 справа)*  **FFFF** |
|  |
| FFFF0 |
| ***и к полученному числу прибавляется компонента смещения (иначе - относительный адрес).*** |

В качестве смещения выбирается ***исполнительный адрес* EA - *E*xecutive address**, который формируется в соответствии с заданным режимом (способом) адресации или просто содержимое специальных регистров CPU (IP,SP,SI, DI). Таким образом, физический адрес **PA**  мы получаем по формуле:

|  |  |  |
| --- | --- | --- |
| **PA=16\*SEG + EA** | | |
| G | | |
| Физический адрес **(20**-ти разрядный) | Сегментный адрес | Смещение |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Сегмент (**Seg**) | | 0 | 0 | 0 | 0 |
| 19 | 4 | 3 | 2 | 1 | 0 |
|  | | | | | |



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **0** | **0** | **0** | **0** | Смещение | |
| 19 | 18 | 17 | 16 | 15 | 1 |
|  |  | C:\Users\Pushkin\AppData\Local\Microsoft\Windows\INetCache\Content.Word\в.png | | | |
|  |  |
|  |  | Суммирование со сдвигом | | | |
|  |  |  |  |  | |
|  |  |  |  |  | |
|  |  |  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Физ Физический адрес **PA ⇒** | **20**-ти разрядный физический адрес | | **⇒** Поступает на **20**-ти разрядную ША |
|  | 19 | 0 |  |

Итак, каждому сегменту соответствует некоторая непрерывная область памяти в **64** Кб, а ***относительный адрес*** – это действительное смещение от начала этой области памяти. Исходя из этого, для записи адреса используется следующая форма:

|  |  |
| --- | --- |
| **AS : AO – offset (**смещение**)** | |
| **з** | |
|  | |
| Сегментный адрес | Смещение |

***Пример:*** дан логический адрес **4F00h : 0010h,** найдем соответствующий ему физический

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | Физический адрес | Сегментный адрес | Смещение |
|  | |  | | |
| адрес | **PA** | **= 4F00h \* 10** | **+ 0010h** | **= 4F010h** |

Исполнительный адрес (***эффективный адрес***) может быть константой, содержимым регистра, содержимым ячейки памяти или суммой нескольких величин, но это число **всегда** **16**-ти разрядное, возникающий при суммировании перенос (переполнение) игнорируется.

Физический адрес никогда не перейдет границу в **1** Мб. Если к последнему физическому адресу **FFFFFh** добавить **1**, произойдет полное обнуление **20**-ти разрядов и *адресное пространство как бы* ***свернется в кольцо***, отсчет адресов начнется сначала:

|  |  |
| --- | --- |
| + | FFFFFh |
| **1** |
| **1|00000h** | |

Исполнительный адрес, в свою очередь, никогда не перейдет границу **64**-х Кб сегмента, на начало которого указывает содержимое сегментного регистра. Сегмент также получается ***свернутым в кольцо*** – по мере увеличения исполнительный адрес достигает значения **FFFF**h (64Кбайт – 1), снова обнуляется и начинает расти сначала. (Сворачивание сегментов и всего адресного пространства в кольцо в МП 80086/88 никак не контролируется.) С одной стороны, это свойство обеспечивает некоторую защиту сегментов друг от друга, но с другой стороны, сегментация памяти является неудобством при написании больших программных модулей. Достоинство сегментной организации памяти: модульность программ – четко выражены области данных, стека и инструкций, а в защищенном режиме – возможность организации многозадачности.

Сегментация адреса обеспечивается при любом режиме работы CPU. Она реализуется путем разбиения адресного пространства на блоки (сегменты), доступ к которым производится в соответствии с определенными правилами. Сегменты хранятся с определенными атрибутами: расположение, размер, тип (стек, команда, данные).

Итак, **для обращения к ячейке сегментированной памяти используется** ***составной (ЛОГИЧЕСКИЙ) адрес* AS:AO.**

Преобразование логических адресов в физические всегда однозначно, т. е.  
паре **АSeg:ЕА** отвечает единственный физический адрес. Обратное преобразование не является однозначным: физический адрес можно представить с помощью 64 Кб/16=4096 пар логических адресов.

Для лучшего понимания можно сравнить сегментную адресацию с поиском адреса человека. Если мы ищем дом в городе Гродно, то нужно знать две составляющие:

*название улицы****:*** *номер дома (смещение от начала улицы)*

Итак, у адреса 2 составляющие. Разберемся далее с каждой из них. Первая составляющая — номер сегмента оперативной памяти (ОП).

Дадим несколько определений:

***Параграфом*** называется фрагмент ОП длиной в 16 байтов, начинающийся с адреса, кратного 16 (10h).

***Сегментом*** ОП называется ее фрагмент длиной в 64 Кб, начинающийся с *параграфа*.

Таким образом, память вся покрыта частично покрывающими друг друга сегментами длиной 64Кб, начинающимися с начала параграфа (см. рис.1.6).

При загрузке программы с диска в ОП программные сегменты записываются в сегменты ОП. При этом адрес сегмента в ОП автоматически записывается в соответствующий сегментный регистр (без последнего 0h) и называется ***сегментным адресом***. Сегментный адрес занимает 16 двоичных разрядов. Зачем адреса сегментов записываются в сегментные регистры? Для того чтобы процессор знал, где у него находятся сегменты исполняемой программы, и мог с ними работать.

0

***Segment 0***

***Segment 1***

DS: 0000

DS:FFFFh

CS: 0000

CS: FFFFh

= 65535*б*.

= 65551*б*.

16=10h

**…………………………………………………………………….**

**Рис.** Сегментная адресация

**1** — базовый адрес сегмента; **2** — смещение; **3** — физический адрес конца сегмента

***Смещением объекта*** *(команды или данного) в программном сегменте* называется расстояние от объекта до начала сегмента.

**ЗАДАНИЯ ДЛЯ ВЫПОЛНЕНИЯ.**

1. Определить физический адрес памяти, если
   * As = 56FFh, Ao = 00DEh;
   * As = C840h, Ao = 1000h;
   * As = 9ED4h, Ao = 0AB9h;
   * As = 7741h, Ao = C524h;
   * As = 0D85h, Ao = 06BAh.
2. Поставить в соответствие физическим адресам ячеек памяти

а) 5E09Ch; б) 9AF35h: с) AC870h по три различных логических адреса.

1. Исполнительный адрес (Executive Address) операнда высчитывается как сумма трех составляющих. Найдите физический адреса операнда для каждого из трех случаев. Базовый адрес сегмента равен 0E450h.

EA = (86 + A570 + 6020)h;

EA = (DB35 + 00CC + 2AE8)h;

EA = (3981 + 4E90 + A500)h;

1. Рассчитать физический адрес команды в оперативной памяти, если сегмент кода начинается с сегмента оперативной памяти по номеру 232Eh+№ учащегося в группе, а эта команда имеет смещение в сегменте кода, равное 102h―№ в группе. Чему равно содержимое регистров CS и IP, если известно, что эта команда выполняется в настоящий момент и ее длина равна 3 байтам?
2. Переменная находится в сегменте данных. Ее физический адрес в оперативной памяти после загрузки программы оказался 51FFAh+№ в группе. Сегмент данных начинается с сегмента оперативной памяти с но­мером 51FEh―№ в группе. Каково смещение переменной относительно начала сегмента?
3. Данные программы имеют длину 20байт+номер в группе. Программа состоит из данных и команд. При загрузке в оперативную память оказа­лось, что свободное место начинается по адресу 31B24h―№ в группе. По какому номеру сегмента запишутся данные? По какому — команды?
4. Сегмент стека имеет размер 90 байтов и начинается по адресу 12540h в оперативной памяти. Чему равно значение регистра SP в начале работы программы, а также после записи в стек слова? Каков физический адрес вершины стека в начале работы программы? Запишите этот адрес с помощью сегментной адресации.

**КОНТРОЛЬНЫЕ ВОПРОСЫ.**

1. Дайте определение понятию «физический адрес».
2. Дайте определение понятию «логический адрес».
3. Дайте определение понятию «сегмент».
4. Поясните значение понятия «замкнутость сегмента».
5. Дайте определение понятию «адресное пространство».
6. Поясните значение понятия «замкнутость адресного пространства».
7. Поясните, каким образом происходит формирование 20-битного физического адреса в процессоре.
8. Приведите формулу вычисления физического адреса по известному логическому.
9. Определите размер адресного пространства для CPU i8086.
10. Опишите структуру адресного пространства процессора i8086.
11. Какова разрядность ША CPU i8086?
12. Какова разрядность ШД CPU i8086?
13. Какова разрядность ШД CPU i8088?
14. Назовите начальный и конечный адрес размещения Conventional Memory.
15. Назовите начальный и конечный адрес размещения Upper Memory Area.
16. Назовите начальный и конечный адрес размещения области векторов прерываний.
17. В какой области памяти размещаются файлы ядра операционной системы DOS?
18. В какой области памяти размещаются пользовательские программы?
19. Какие условия накладываются на размещение сегментов в памяти?
20. Доступна ли для CPU i8086 Extended Memory?
21. Каково назначение Expanded Memory?
22. Какие характеристики компьютера зависят от размера ША, ШД?
23. Дайте характеристику CPU i8086 и CPU i8088.
24. Сколько слов помещается в 1 Мб оперативной памяти?
25. Сколько параграфов помещается в 1 Мб оперативной памяти?
26. Что такое сегмент оперативной памяти?. С какого адреса начинается последний полный сегмент оперативной памя­ти до 1 Мб? .Каков номер такого последнего сегмента? Сколько полных сегментов помещается в оперативной памяти до 1 Мб?
27. Почему в сегментный регистр записывают не адрес оперативной памяти, а номер сегмента?
28. Пусть задан физический адрес оперативной памяти 00015h. При сегментной адресации его представляют в виде пары: "сегментный регистр : смещение". Чему будет равно значение сегментного регистра и смещения в данном слу­чае? Приведите хотя бы 2 решения.
29. Физический адрес — 00025h. Условие задачи аналогично предыдущей. Приведите 3 решения.